Network Coding for Non-uniform Demands [1]

Yuval Cassuto
Department of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125, U.S.A.
Email: ycassuto@paradise.caltech.edu

Jehoshua Bruck
Department of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125, U.S.A.
Email: bruck@paradise.caltech.edu

***Abstract*— Non-uniform demand networks are defined as a useful connection model, in between multicasts and general connections. In these networks, each sink demands a certain number of messages, without specifying their identities. We study the solvability of such networks and give a tight bound on the number of sinks for which the min cut condition is sufficient. This sufficiency result is unique to the non-uniform demand model and does not apply to general connection networks. We propose constructions to solve networks at, or slightly below capacity, and investigate the effect large alphabets have on the solvability of such networks. We also show that our efficient constructions are suboptimal when used in networks with more sinks, yet this comes with little surprise considering the fact that the general problem is shown to be NP-hard.**

I. Introduction

Network coding for multicast networks is a well studied, and by now, well understood subject. Starting at the network coding ”big-bang” result of [1] and continuing in a multitude of works, a lot of attention was pointed toward this type of networks, which remain the paramount application for network coding to date. A multicast connection is depicted in figure 1(a). For general connections (figure 1(b)), [2] gives algebraic characterizations of solvable networks, but concludes those are hard to check. Linear network coding was shown to be hard [3] and no alternative coding constructions are known. Other connection models were studied in [2], and for them solvability is equivalent to different combinatorial properties of the network. The model considered in this paper is the model of *nonuniform-demand* networks whose formal definition will follow. In that model, every sink demands a number of information messages from the pool of messages available at the source, as depicted in figure 1(c). It is different from multicast in that a sink demands only a *subset* of the source messages, and it is different from the general connection model since only the *size* of the message subset is specified rather than its exact content. This type of connections is motivated by various applications whereby the rate of communication with the source is important, but the identities of the messages delivered to the different sinks are not. One example is when the source messages are different descriptions of a multi-description code (See [4] for a tutorial and references on multiple description coding). Another is when the source node employs a rateless channel code (e.g Fountain codes [5]).

Definition 1: A *nonuniform-demand network problem* is a directed acyclic graph $\mathcal{G}(V, E)$ (unit capacity edges) with a node s distinguished as the source node, together with a demand function $D : V \rightarrow \mathcal{Z}^*$ (where $\mathcal{Z}^*$ denotes the set of non-negative integers) whose values represent the number of information messages demanded by each node.
The network is said to be *solvable*, if there exists a network code that satisfies all demands simultaneously. An interesting question to ask about nonuniform-demand networks is whether, similarly to multicast networks, solvability can be determined based solely on the minimum cuts between the source and each of the sinks. In section III we pursue a direct generalization of the multicast theorem (min cut of d is both sufficient and necessary for multicast of d units of information) to sinks with different min cuts d_i to the source. We show that such a generalization works when we bound the number of partial demand sinks and that this bound cannot be improved in a general result. We continue to show that further guarantees can be provided when some sinks have min cuts larger than their demand. Also, we argue that for a given network, more can be achieved relative to using the construction used to achieve the worst case guarantees. In section V we discuss the power of large alphabets in the context of nonuniform-demand networks and in VI we show that the general nonuniform-

[1] This work was supported in part by the Caltech Lee Center for Advanced Networking and by NSF grant ANI-0322475

demand problem is NP-hard for linear and nonlinear codes.

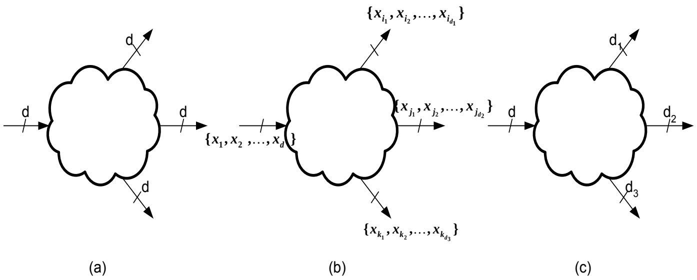
d
d
d
d
$\{x_1, x_2, \ldots, x_d\}$
$\{x_{i_1}, x_{i_2}, \ldots, x_{i_{d_1}}\}$
$\{x_{j_1}, x_{j_2}, \ldots, x_{j_{d_2}}\}$
$\{x_{k_1}, x_{k_2}, \ldots, x_{k_{d_3}}\}$
d
d_1
d_2
d_3
(a)
(b)
(c)

Fig. 1. Illustration of multicast (a), general connections (b), and nonuniform-demand networks (c)

II. Definitions and notations

Let $d_0 = \max_{v \in V} D(v)$, the maximum demand of a sink in the network. Skipping other possible sinks with demand d_0 we number the sink demands d_i, according to a non-increasing order. In other words we have $d_i < d_0$ for all $i > 0$ and for all $0 < i < j$ we have $d_i \geq d_j$ (except for d_0, a d_i value may repeat for different i). Therefore, for $i > 0$, d_i can specify both the sink identity and its demand. For notational compactness we will use both; and the context will make clear the role of d_i in each instance. We will call the sinks d_i for $i > 0$, *partial demand* sinks. Denote by c_0 the min cut between s and any of the sinks with demand d_0. c_i is the min cut between s and the node d_i. Using standard linear network coding terminology, the source emits linear combinations of the information messages on its outgoing edges. The network nodes take linear combinations of the symbols on their incoming edges and output them on their outgoing edges. The collection of these linear combinations is referred to as the network code. The yield of this network code is global coding vectors available at the sinks, which are the resulting linear combinations of the information messages available at the source. In our discussion, we will collect these length d_0 global coding (row) vectors into $d_i \times d_0$ matrices. For an $m \times n$ matrix $A = [\overline{a}_1, \overline{a}_2, \ldots, \overline{a}_n]$, define the zero column index set $Z_A = \{i : \overline{a}_i = \mathbf{0}\}$. Also define the nonzero column index set $N_A = \{1, \ldots, n\} - Z_A$. For a set of indices T, define $A(T)$ to be the submatrix of A that consists of the columns T. We say that a rectangular $l \times n$ matrix A is invertible $(n > l)$ when $|Z_A| = n - l$ and $A(N_A)$ is an invertible $l \times l$ matrix.

Definition 2: A (elementary) column operation on a matrix B is called Z_A*-contaminating* if it adds a nonzero multiple of a column in N_A to a column in Z_A. A column operation which is not Z_A-contaminating will be called Z_A*-non contaminating*.

III. Generalization of multicast networks

A. Two partial demand sinks

The following theorem gives the best possible generalization of the multicast capacity to the nonuniform-demand problem.

Theorem 3: A nonuniform-demand connection with m sinks, each with demand d_0 and 2 sinks with demands $d_1 < d_0, d_2 < d_0$, is solvable using linear codes if and only if the minimum cut between the source and each of the $m + 2$ sinks is greater or equal to its demand.

For the case of $m = 1$ we get the following corollary.

Corollary 4: Any three demands in a network can be simultaneously satisfied using linear codes if and only if the minimum cut between the source and each of the sinks is greater or equal to its demand.

Proof of theorem: We prove the sufficiency part. Assume $c_i = d_i$ for all i. We show that taking a linear multicast code for the m users with demands d_0 (such a code exists and can be found in polynomial time [6],[7]), one can transform it into a code for all $m + 2$ users using linear operations on the information symbols. The proof will use the following argument. If the network code encodes a message vector u and provides a sink with a matrix M of global coding vectors, and if that sink prefers having a matrix $G = MP$ instead (for any invertible P), then the source that has access to all messages can encode the vector $u' = Pu$ and provide the sink with that desired G. Of course all the sinks' global coding vectors will change appropriately. The code construction algorithm of [7] can be directly extended to provide each sink with $c_i = d_i$ linear independent global coding vectors of length d_0. Note that in general $c_i \leq d_0$ so that stage alone only guarantees the satisfaction of the d_0 demands. Group the d_1 coding vectors for user d_1 into a $d_1 \times d_0$ matrix A. Similarly, group the d_2 coding vectors for user d_2 into a $d_2 \times d_0$ matrix B. Using Gaussian elimination on the columns of A we get $A^{(1)} = AP^{(1)}$, where $P^{(1)}$ is a $d_0 \times d_0$ invertible matrix and so $A^{(1)}$ has exactly d_1 nonzero columns and rank d_1. The matrix $A^{(1)}$ is thus invertible and d_1 information messages can be recovered at the sink. We now show that a similar invertible matrix can be obtained for user 2 using $Z_{A^{(1)}}$-non contaminating column operations thus maintaining the zero columns of $A^{(1)}$ along the process. First we define $B^{(1)} = BP^{(1)}$. Let $r = \operatorname{rank}\left(B^{(1)}(Z_{A^{(1)}})\right)$. We distinguish two cases.

Case I: $r = d_2$

If $B^{(1)}(Z_{A^{(1)}})$ has a maximal rank, a Gaussian elimination can be carried out with only $Z_{A^{(1)}}$-non contaminating column operations, leaving exactly $d_0 - d_2$ all zero columns.

Case II: $r < d_2$

In this case Gaussian elimination can contaminate columns in $Z_{A^{(1)}}$ since column exchanges may be necessary between $Z_{A^{(1)}}$ and $N_{A^{(1)}}$. We will then use the following process. Using only column operations on $B^{(1)}(Z_{A^{(1)}})$ we can zero $|Z_{A^{(1)}}| - r$ of its columns. By adding multiples of columns from $B^{(1)}(Z_{A^{(1)}})$ to columns in $B^{(1)}(N_{A^{(1)}})$ we can zero r **rows** of $B^{(1)}(N_{A^{(1)}})$. Denote this modification of submatrix $B^{(1)}(N_{A^{(1)}})$ by $\tilde{B}$. That ensures that

$$\mathrm{rank}(\tilde{B}) \leq d_2 - r$$

Now Gaussian elimination of $\tilde{B}$ alone using column operations can zero $|N_{A^{(1)}}| - \mathrm{rank}(\tilde{B})$ of its columns. The aggregate column operations performed on $B^{(1)}$ result in a matrix $B^{(2)} = B^{(1)}P^{(2)}$, where $P^{(2)}$ is invertible and includes only $Z_{A^{(1)}}$-non contaminating column operations. We can sum up the zero columns of $B^{(2)}$ and get

$$|Z_{B^{(2)}}| = (|Z_{A^{(1)}}| - r) + (|N_{A^{(1)}}| - \mathrm{rank}(\tilde{B})) \geq$$

$$\geq |Z_{A^{(1)}}| - r + |N_{A^{(1)}}| - d_2 + r = d_0 - d_2$$

Since $\mathrm{rank}(B) = d_2$, using invertible column operations we need have $|Z_{B^{(2)}}| \leq d_0 - d_2$ so we finally get

$$|Z_{B^{(2)}}| = d_0 - d_2$$

Therefore both $B^{(2)}$ and $A^{(2)}$ are invertible. □

Note that the *identities* of the messages received by d_2 cannot be determined freely. Those may depend on the network and the specific code the construction started with. Therefore, this result is unique to the nonuniform-demand case as defined above. We next use a network example to prove that the above construction cannot be improved, in general.

Theorem 5: There exist unsolvable nonuniform-demand networks that consist of 3 partial demands and all its demands satisfy $d_i \leq c_i$.

Proof: Consider the example in figure 2 (assume edges are pointed downward). Demands of 1 for d_2 and d_3 force the source to emit pure symbols on both its outgoing edges. The additional demand of d_1 disallows node C to perform coding. Without coding, at least one of the demands d_0, d_0' cannot be satisfied. We conclude that the network is not solvable. □

Theorem 3 shows that any nonuniform-demand network with at most two partial demands is solvable. The example in theorem 5 shows that the number of solvable nonuniform demands cannot exceed two, in general. Note that the source interference cancellation that was used to attain capacity in theorem 3 is weak in the sense that it takes *any* linear multicast code for the d_0 demand sinks, and transforms it into a network code for all the sinks, using only linear operations on the source symbols. It is thus surprising that this construction guarantees the maximum possible partial demand sinks in a nonuniform-demand problem.

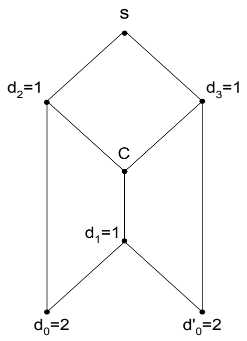

Fig. 2. Unsolvable connection with three partial demands

B. More than two partial demand sinks

Theorem 5 deems impossible any attempt to provide further guarantees for nonuniform-demands that equal the min cuts. However, more guarantees can be obtained once we operate a subset of the sinks below their best-wished capacity ($d_i < c_i$). This concept of achieving more by relaxing the strict capacity requirement has already appeared in the network coding literature, and was found significantly useful [8],[9]. Here the surprising fact is that even a small number of sinks having more linearly independent coding vectors than they need can help satisfying all simultaneous demands. As one instance of that idea, we shall state (without proof) a theorem concerning networks with 3 partial-demand sinks.

Theorem 6: A nonuniform-demand connection with m sinks, each with demand d_0 and 3 sinks with demands $d_3 \leq d_2 \leq d_1 < d_0$, is solvable using linear codes if $c_i = d_i$ for $i = \{0, 1, 2\}$ and $c_3 = d_3 + (d_0 - d_1)$.

As an example, theorem 6 guarantees solvability of any network with demand vector $\bar{d} = [6, 6, 5, 4, 3]$ if the corresponding min cut vector is element-wise at least $\bar{c} = [6, 6, 5, 4, 4]$. Theorem 6 requires sink d_3, the smallest demand sink, to have a min cut larger than its demand. However, this is a special case and in the more general one, other sinks can be required to have large min cuts. Moreover, this method can be generalized to more than 3 partial demand sinks, though formulating the exact guarantees becomes tedious when increasing the number of sinks.

IV. Suboptimality of source cancellation

Although we showed that the source interference cancellation used in the proof to theorem 3 is optimal for general networks, for a particular network using it may render suboptimal results. Indeed, for networks with more than 2 partial demand sinks, more can be achieved by not limiting the interference cancellation to the source only. We will show this using the network in figure 3. In figure 3(a), an extension of the multicast code construction algorithm is run, providing each sink with d_i linearly independent vectors of length 3. The two independent vectors obtained by sink d_1 are $\left[\begin{array}{ccc}0&0&1\\\hline 1&1&1\end{array}\right]$ (these correspond to C and $A+B+C$ shown in figure 3(a)). It is easy to see that it is impossible to make this 2×3 matrix invertible using column operations that are non-contaminating for both d_2 and d_3; adding column 1 to column 2 would contaminate sink d_2 and adding column 2 to column 1 would contaminate sink d_3. That means this network code cannot be modified to satisfy the demand of d_1 with only source operations. In figure 3(b), on the other hand, a network code is given that satisfies all demands.

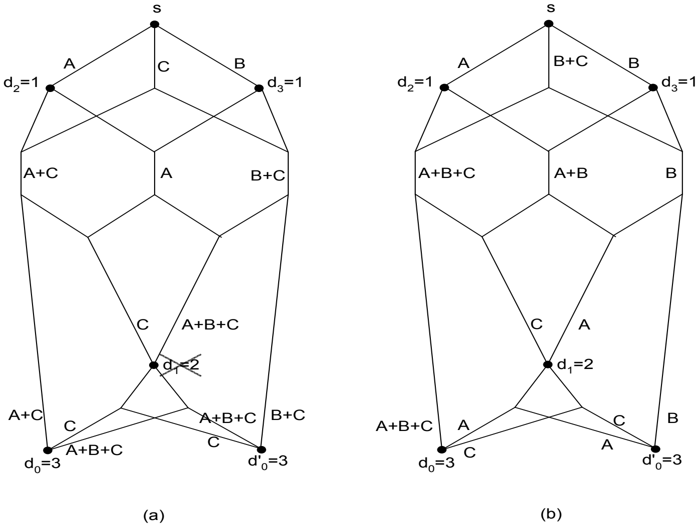

Fig. 3. Code modification (a) fails in achieving network capacity achieved in (b).

V. The power of large alphabets

Many known results connect the solvability of networks to the alphabet size used for coding [7],[2],[10],[3] (and more). In this section we seek to contribute to the above discussion, considering nonuniform-demand networks. We will show that for nonuniform-demand networks, alphabets which are sufficient to providing d_i linear independent vectors to each sink, are insufficient to solving the nonuniform-demand problem. That implies that the upper bound of $|T|$ (the number of sinks), given in [7] for the required field size, may not apply to nonuniform-demand solvable networks. For a given network, define $\mathcal{Q}_{LI}$ to be the smallest field size capable of providing d_i linear independent global coding vectors to each sink. Define $\mathcal{Q}_{ND}$ to be the smallest field size capable of satisfying the nonuniform demands d_i in each sink. Since having d_i linear independent vectors is a necessary condition to satisfying the demands, we have $\mathcal{Q}_{LI} \leq \mathcal{Q}_{ND}$. Beyond satisfying that necessary condition, it is unclear whether large alphabets can solve nonuniform-demand networks that are unsolvable using smaller alphabets. The following theorem answers this question to the affirmative.

Theorem 7: There exist networks where $\mathcal{Q}_{LI} < \mathcal{Q}_{ND}$.

Proof: We will show a network for which a binary alphabet is sufficient to provide each sink with d_i linear independent vectors, but insufficient to solve the network. The network will be shown to be solvable using larger alphabets (e.g $GF(3)$). For a binary network code, we can force an edge to carry the symbol $X+Y$ using the gadget given in figure 4. We will use such gadgets to obtain the network in figure 5. Using a binary code, edge e can carry an element from the set $\{0\ ,\ B+C\ ,\ B+D\ ,\ C+D\}$ or an element from the set $\{A+B\ ,\ A+C\ ,\ A+D\ ,\ A+B+C+D\}$. An element from the first set will provide at least one of the sinks with linearly dependent vectors. An element from the second set will provide each sink with 3 linearly independent vectors, but none of them will result in a valid solution. Taking the message symbols to be from the field $GF(3)$, edge e can carry $(A+B)+(A+C)+(A+D) = (B+C+D) \bmod 3$. This combination allows both sinks to obtain messages B, C, D. □

We remark that the network used in the proof is solvable using any alphabet of size $q > 2$. Yet it is unknown whether this threshold effect is true in general.

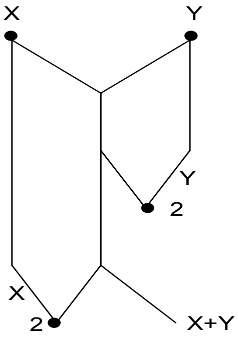

Fig. 4. Gadget to force $X+Y$ in a binary network code

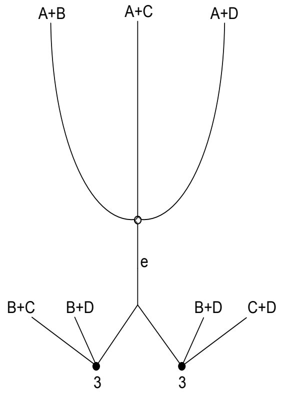

Fig. 5. Solvable network where alphabet of size 3 is required

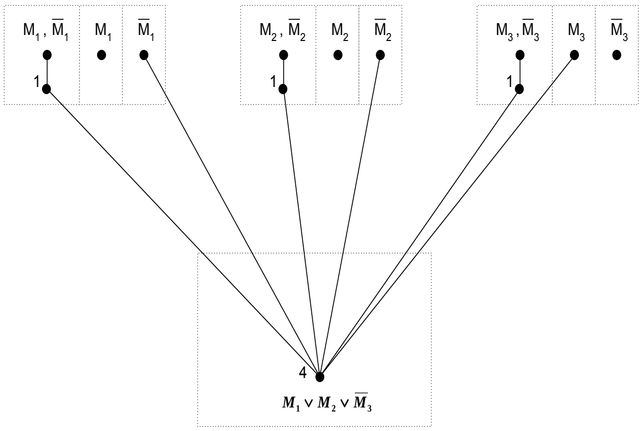

Fig. 6. network corresponding to the clause $X_1 \vee X_2 \vee \bar{X}_3$

VI. Hardness of the nonuniform-demand problem

In [3] it has been shown that it is hard to find linear codes for connections where sinks demand arbitrary sets of information messages. However, this result does not establish the hardness of the nonuniform-demand case since [3]'s reduction assumes sinks demand *specific* messages while in the nonuniform-demand problem only a *number* of messages can be demanded. Evidently, specifying just the number of messages demanded by a sink imposes a milder objective for the network code, compared with the specific messages case. However, as it turns out, the general nonuniform-demand problem is NP-hard as well, even when not restricted to linear codes. To show that, we will use a simple reduction from 3-SAT. Given a 3-CNF formula f over $X_1, \ldots, X_n$, we construct a corresponding nonuniform-demand network. For every variable x_j we define a gadget consisting of three sources. One has access to M_j, one to $\bar{M}_j$ and another one to both $M_j, \bar{M}_j$. In addition, the variable gadget has a node with demand of 1, connected to the source with access to both $M_j, \bar{M}_j$ (see figure 6). For a clause $X_1 \vee X_2 \vee \bar{X}_3$ we define a clause gadget with a single sink connected as shown in figure 6. We assign a demand of 4 to the node in each clause gadget.

Proposition 8: f is satisfiable if and only if the corresponding nonuniform-demand network is solvable.

Proof:
($\Rightarrow$) Let $X = (x_1, x_2, \ldots, x_n)$ be a satisfying assignment of f. Then each of the left nodes in each variable gadget can output M_j if $x_j = 1$ and $\bar{M}_j$ if $x_j = 0$. In that case it is obvious that all the clause sinks will be able to recover at least 4 messages.
($\Leftarrow$) If the network is solvable each of the demand 1 sinks has exactly one of M_j and $\bar{M}_j$. We can assign $x_j = 1$ if it has message M_j and $x_j = 0$ if it has $\bar{M}_j$. This assignment satisfies f since in every clause at least one of the variables contributes a message not obtained using the direct links, and this message is consistent across all clauses.□

VII. Conclusion

Non-uniform demand networks were proposed as a useful model of network connections. Some sufficiency results that are unique to that model were proved and other insights pertaining to such networks were given. Still the study of non-uniform demand networks is only at its initial steps. A more complete characterization of solvable/unsolvable networks is still missing and stronger algorithms to design codes are needed.

References

[1] R. Ahlswede, N. Cai, S.-Y.R Li, and R.W. Yeung. Network information flow. *IEEE-Trans-IT*, 46(4):1204–1216, July 2000.

[2] R. Koetter and M. Medard. An algebraic approach to network coding. *IEEE/ACM-Trans-Networking*, 11(5):782–795, October 2003.

[3] A.R Lehman and E. Lehman. Complexity classification of network information flow problems. In *proc. of the forty-first Allerton conference on control, communication, and computing*, Allerton, IL, October 2003.

[4] V. Goyal. Multiple description coding: Compression meets the network. *IEEE-Signal Processing Magazine*, pages 74–93, September 2001.

[5] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, and V. Stenmann. Practical loss-resilient codes. In *proc. of the Symposium on Theory of Computing*. ACM, 1997.

[6] S.-Y.R Li, R.W. Yeung, and N. Cai. Linear network coding. *IEEE-Trans-IT*, 49(2):371–381, February 2003.

[7] S. Jaggi, P. Sanders, P. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen. Polynomial time algorithms for multicast network code construction. *IEEE-Trans-IT*, 51(6):1973–1982, June 2005.

[8] A.R Lehman and E. Lehman. Network coding: does the model need tuning? In *proc. of the 16th Symposium on Discrete Algorithms*, January 2005.

[9] J. Cannons, R. Dougherty, C. Freiling, and K. Zeger. Network routing capacity. *Submitted to IEEE/ACM-Trans-Networking, October 2004*.

[10] R. Dougherty, C. Freiling, and K. Zeger. Linearity and solvability in multicast networks. In *proc. of the conference on information sciences and systems*, Princeton university, March 2004.